Sci.Int.(Lahore),26(1),97-100,2014 ISSN 1013-5316; CODEN: SINTE 8

A UNIVERSAL VIEW SOA INTEROPERABILITY FRAMEWORK FOR MULTIPLE C4I APPLICATIONS

Zeeshan Siddiqui, Abdullah S. Alghamdi
C4I Centre of Advance Systems (C4ICAS), King Saud University
Riyadh, Kingdom of Saudi Arabia
[zeeshan.siddiqui, ghamd]@ksu.edu.sa

ABSTRACT:*This study proposes a Universal View (UV) interoperability framework based on service oriented architectures (SOA). The proposed framework is intended to interoperate multiple C4I instances over a loosely coupled SOA architecture. In this paper we have discussed the general working of our UV framework; the core testing and implementation will be discussed in further extended studies.*

Keywords— Service Oriented Architecture (SOA), Middleware, Interoperability, C4I

I. INTRODUCTION

Interoperability is the most discussing topic these days as organizations are growing and communication is taking place. Different integration and communication techniques based on different approaches were presented and implemented and many approaches are yet to be proposed. The contemporary technology of this era is enhancing, this enhancement is moving the requirement of architectures to most discreet level. The interoperability between different applications and softwares working under identical policies and proposals is still a challenge, depending on the size of their organizations. When there is a case of applications based on different platforms needs to communicate to each other under a unique common field, the interoperability issue turn towards a more complex and difficult situation. [1]

In this study, our focus is not a business oriented organization, rather, a defense based organizations. All of these military organizations are having their own Command Control Computer Communication & Intelligence (C4I) architectures, based on the platform customized according to their need and requirements. [1]

In a war like situation, mission target achievements depend on command, control, communication and reporting. More over, it is mandatory to accurately understand or receive what is sent from the command and control authorities or specified commanders. Military channels should be the most secured and threat resistant networks to avoid hack and attacks, because, any disclosure of communication can result in a complete failure of that mission specific task. Therefore, security is another important aspect that can affect interoperability ability of any framework.

A military organization comprises of hundreds of applications running, that may be customized according to the organization need or bought from a third-party vendor or can be a part of an inherited system. Number of application is not the case; the infrastructure should be capable of exchanging applications and absorbing new scenarios. The term basically used in this aspect is Enterprise Application Integration (EAI) [2].

The Message-Oriented Middleware is the traditional solution to EAI, the commodities build using MOM comprises of a central message line up system frequently termed as message broker. Applications are connected to the message broker to put message in and out. A unique capability of the message broker is to store messages, in a way that dispatcher and recipient do not have to be logged in at the same time. Message broker can route the message to transmit it to more then one receiver and it can help the receiver to fit the message according to its own requirement. This facility permit the connected applications to use own required message formats. [3]

One of the big interoperability issue of MOM is the platform dependency, protocols and platforms are specific not independent. Requirement of a reusable service that is independent of any specific platform and that can help implement the complex business processes is mandatory. Service Oriented Architecture (SOA) [4] is the modern and efficient use of this phenomena; SOA can be build and put into practice by several web service technologies i.e. SOAP [5] or REST [6]. With the usage of SOA in identical platforms, it also gives benefit to those enterprise application providers based on dissimilar OS platforms and using different programming languages. Due to these facts, a Universal View (UV) should be present acting as a middleware that map or re-map the application to grant it access and integration. In this paper we propose a SOA based interoperability UV that allows the application to map/remap regardless of platform boundaries.

Rest of the paper formation is as follows; in Section II we discuss literature review with respect to middleware architectures and C4I application integration, in Section III we discuss the functionality of our propose UV, in Section IV we proposes our architecture framework, Section V cover conclusion, future work and final remarks.

II. LITERATURE REVIEW

In this section we streamline few studies among many researches that have proposed modelled or implemented SOA based middleware architectures in military and non military domains.

Nenad M. et al discussed model of data exchange between heterogeneous systems using Mule enterprise service bus. They implemented Mule ESB on Information System for Higher Education (ISVU) and Student's Nourishment Information System (ISSP). [7]

Pooia L. et al in their paper focuses on the role of service oriented architectures in situational awareness and how it can help generating a more extensive and comprehensive situational awareness. [8]

Peter B. et al in their study focuses on architecture and technical implementation of semantic service buses and other core modules of the SPIKE system for business collaboration. The interoperability of heterogeneous applications and services provided by the alliance partners is achieved by employing the semantic technologies, namely the WSMO and Sa-WSDL extensions. [9]

Alghamdi, A and Siddiqui, Z. proposed a common interoperability framework between defense architectures. The model integrate the application architecture sensitive data in the form of RDF files on a secure and responsive security channel which is then picked up by the web crawlers. The data resources are then gathered and collected in the form of meta-data repository. [10]

Paul S. et al, presented a cross domain information Exchange Framework (CIEF). This architectural framework is designed to support critical information exchange toassist or automate DoD (Department of Defense) mission oriented tasks. It is also an operational design for the publication, location, and subscription to information in the correct mission context and monitors the operational use of information in that context. [11]

Boggs. D. outlined the technical issues in SOA in a bandwidth limited environment. Additionally, the paper presents an approach to engineer solutions for this environment, identifies promising techniques specific to the technical problems, and proposes research to further potential solutions for using an SOA in constrained conditions. [12]

III. PROPOSED UNIVERSAL VIEW (UV) FUNCTIONALITY

Our proposed UV framework orbits around SOA, it serves as a middleware that can assist interoperability of assorted services and applications in an enterprise. By defining standardized service interfaces and message routing, the UV negotiate incompatibilities of communicating applications, arrange their interactions, and build the integrated services accessible for wide access and re-use. Technically speaking, the UV permits loose coupling of interrelated systems and enables to allocate the business logic of a solution into digestible modules, maintaining their own local control and independence. Our UV provide following SOA based functionalities:

- Routing messages between services
- Converting transport protocols between requester and services
- Transformation of message formats between requester and services
- Handling of business events from dissimilar sources

A. UV Cross-Domain Functionality

UV is designed specifically to call upon services without locating them and without know the method to invoke them. The application developer is able to focus on defining a nonspecific application message layout, and leaving the protocol transformations to be administered by the UV.

Developer does not need to focus on the dissimilarities based on protocols but can rely on the UV to convert and route the request to a provider that can respond to the request.

The purpose of UV is to give an ideal hold to implement the service oriented architectures, without conciliation to protection, scalability, consistency and performance.

A. Working

For inter-application messaging UV is a fundamental component that provides a loosely coupled framework. Message delivery is supported by UV server by multiple protocol bindings, such as WSIF, HTTP/SOAP, JCA, JMS and java, by utilizing asynchronous/synchronous, reply/request, subscribe/publish models.

A.1 UV Interoperation (UVI)

3 tired approach is a common but most essential architecture approach these days, UV fully understand and support all three layers i.e. User Interface (UI) Tier, Middle Tier and Data Tier.

UVI is the combination of Business rules and transformation logic. These rules and logics help to integrate heterogenous applications. By modelling the integration logic, result is metadata that is stored in the UVI repository. In integration design, during runtime, the metadata is treated as runtime instructions to exchange the conversation between applications.

Application that are communicating and exchanging information uses the UV; there is no direct communication between applications. If any change or upgrade in the linked application occurred, then the changes reflects in that particular application view and maps to the UV. In other words, only remapping is required, rest of the relationships with the hub remains unaffected

IV. PROPOSED UV FRAMEWORK

On the basis of our discussion in Section III, we now put forward our SOA based interoperability framework. The main purpose of proposing UV is to overcome the most occurring interoperability issues. When communication is spaning and exchanging information on a common battle field between different military C4I systems having different platforms and standards, the interoperability issues become a painstaking exercise. We have summarized Interoperability issues with respect to SOA as follows:

- Lack of multi-protocol bus dependency
- Virtualization
- Transformation
- Routing
- Availability/Scalability
- Open standards

Related to our discussion in the above section, our framework architecture fully covers the foresaid issues, in a professional and monitored way. Following Table 1.0 demonstrate the interoperability issues and UV's functionality to overcome these issues:

Multi-protocol issues are tackled by separating integration concerns from applications and business logic. Overcome **Virtualization** by connecting to target using web services,

Sci.Int.(Lahore),26(1),97-100,2014 ISSN 1013-5316; CODEN: SINTE 8

applications, legacy, technologies and non web service connectivity, through adapters. Overcome **Transformation** by converting data as it flows through the bus.
Overcome **Routing** issues through reliably transport and route data over a variety of protocols. Achieve **High-availability** by introducing highly available topologies and implementing high availability on 3-tier architecture. Overcome **Open Standards** issues by introducing **standard drive hot plug ability** through out **virtualization, transformation and routing.**

TABLE I. INTEROPERABILITY CRITICAL ISSUES & UV FUCTIONALITY

Interoperability Issues	UV Functionalities
Multi-protocol bus dependency	Separate integration from applications and business logics
Virtualization	Usage of web service and non-web service connectivity through adapters
Transformation	Data conversion as it flows
Routing	Reliable data transportation over protocols
Availability	3-tier architecture
Open Standards	Drive Hot Plug ability

By briefly discussing features and functionalities of **UV** in a summarized manner, we can integrate the different C4I applications in a way that does not change the current architecture of any application. The only option we have is to connect all applications to a universal view. **UV** does this job efficiently working as a SOA based middleware architecture. The below model demonstrates the connectivity between multiple C4I instances:

CONCLUSION & FUTURE WORK

Our SOA Interoperability Framework is providing complete and comprehensive integration to different domains related to different organizations following different architectures. It gives a far more ahead of approach that overcomes the traditional hard-coded integration methods. This framework focuses on functional aspects of integration, giving absolute platform independency. It does not only provide ***platform-to-platform*** integration, but, by using pre-packed adapters, focuses more on ***cross-platform*** integration techniques and methodologies. The following conclusions are lined up to overcome ***interoperability*** issues with respect to SOA:

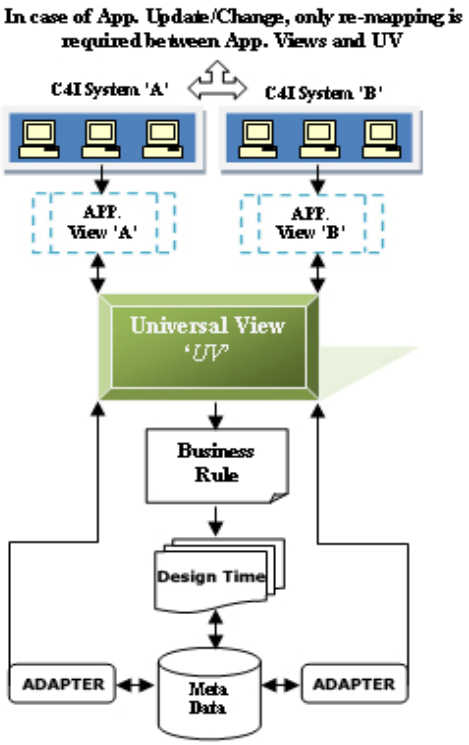

Figure 4.1, proposed integrated model using *OracleAS Interconnect*

- ***Loose Coupling,*** by giving a universal view not a direct connection to decrease integration interfaces count.
- ***Ease of Customization,*** any sudden or certain change in any application will not affect the universal view; rather, require a re-map to the UV.
- ***Easy Expansion,*** the integration of any new application in the system is as easy as *plug-n-play.* The particular application view needs to be connected to the common-view and any of this addition will not affect the existing applications connectivity.

In our forthcoming research papers we are planning to discuss the implementation and testing of our proposed framework by implementing and *put-into-play* our architecture framework to come up with accurate results and outcomes that strongly supports our discussion in this paper.

REFERENCES

[1] Siddiqui. Z., Ghamdi. A., *"Node level Information Security in Common Information Exchange Model (CIEM)"*, Science International, ISSN 1013-5316, December 2010

[2] AlGhamdi. S, Siddiqui. Z and Quadri. S, *A Common Information Exchange Model for Multiple C4I Architectures, 2010 UKsim 12th IEEE International Conference on Computer Modelling and Simulation*, 24-26 March 2010.

[3] Linthicum. D., *Enterprise Application Integration*, Addison-Wesley Longman Ltd. Essex UK, UK, Published in 2000.

[4] Li. H. and Jiang. G., *Semantic Message Oriented Middleware for Published/Subscribed Networks,* proc of SPIE, 2004 – actcomm.thayer.dartmouth.edu

[5] Agrawal. R., Bayardo. R., Gruhl. D., Papadimitriou. S., *Vinci: A SOA for rapid development of web applications,* Elsevier Science, march 16 2001

[6] W3C, *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition),* W3C Recommendations, April 27, 2007

[7] Fielding. R. and Taylor. R., *Principled Design of the Modern Web Architecture,* ACM Transaction on Internet Technology, Vol. 2, No. 2, Page 116 of 115-150

[8] Nenad Milanovic, Tanja Milicevic, *Model of data exchange between heterogeneous systems using Mule enterprise service bus* , Proceedings of the *ITI 2009 31st Int. Conf. on Information Technology Interfaces,* June 22-25, 2009, Cavtat, Croatia

[9] Pooia Lalbakhsh, Nasrin Sohrabi, Mehdi N. Fesharaki, *Role of SOA in Situation Awareness,* 2nd IEEE International Conference on Computer Science and Information Technology, Beijing, China August 08-August 11, ISBN: 978-1-4244-4519-6

[10] Peter BEDNÁR a, Karol FURDÍK b, Gabriel LUKÁČ a and Tomáš SABOL, *Design of a Semantic Service Bus for Networked Enterprises*, AMIF 2009 - Ambient Intelligence Forum 2009. Hradec Králové, Czech Republic, September 16-17, 2009

[11] Abdullah S. Alghamdi and Zeeshan Siddiqui, *Common Information Framework b/w Defense Architecture, A Web Semantics approach*, 16th International Conference on Distributed Multimedia Systems, Hyatt Lodge at McDonald's Campus, Oak Brook, Illinois, October 14-16, pp. 149-152, 2010 USA

[12] Paul Shaw, Dr. David J. Roberts**,** *The Cross-domain Information Exchange Framework (CIEF),* AFCEA-GMU C4I Center Symposium "Critical Issues in C4I" GMU's Fairfax, Virginia Campus 20-21 May 2008

[13] J. D. Boggs, *Framework for an SOA in Bandwidth-Limited Environments*, AFCEA-GMU C4I Center Symposium "Critical Issues in C4I" GMU's Fairfax, Virginia Campus 20-21 May 2008